An Image Encryption Scheme Based on Concatenated Torus Automorphisms

Qian Mao[1, 2], Chin-Chen Chang[2, 3] and Hsiao-Ling Wu[3]
[1] Department of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology
Shanghai, 200093, P. R. China
[e-mail: maoqiansh@gmail.com]
[2] Department of Computer Science and Information Engineering, Asia University
Taichung, 41354, Taiwan
[e-mail: alan3c@gmail.com]
[3] Department of Information Engineering and Computer Science, Feng Chia University
Taichung, 41354, Taiwan
[e-mail: wuhsiaoling590@gmail.com]
*Corresponding author: Chin-Chen Chang

Received January 8, 2013; revised March 18, 2013; revised April 29, 2013; accepted May 18, 2013; published June 26, 2013

Abstract

A novel, chaotic map that is based on concatenated torus automorphisms is proposed in this paper. As we know, cat map, which is based on torus automorphism, is highly chaotic and is often used to encrypt information. But cat map is periodic, which decreases the security of the cryptosystem. In this paper, we propose a novel chaotic map that concatenates several torus automorphisms. The concatenated mechanism provides stronger chaos and larger key space for the cryptosystem. It is proven that the period of the concatenated torus automorphisms is the total sum of each one's period. By this means, the period of the novel automorphism is increased extremely. Based on the novel, concatenated torus automorphisms, two application schemes in image encryption are proposed, i.e., 2D and 3D concatenated chaotic maps. In these schemes, both the scrambling matrices and the iteration numbers act as secret keys. Security analysis shows that the proposed, concatenated, chaotic maps have strong chaos and they are very sensitive to the secret keys. By means of concatenating several torus automorphisms, the key space of the proposed cryptosystem can be expanded to 2^{135}. The diffusion function in the proposed scheme changes the gray values of the transferred pixels, which makes the periodicity of the concatenated torus automorphisms disappeared. Therefore, the proposed cryptosystem has high security and they can resist the brute-force attacks and the differential attacks efficiently. The diffusing speed of the proposed scheme is higher, and the computational complexity is lower, compared with the existing methods.

Keywords: Image encryption, chaotic map, torus automorphism, permutation, diffusion

http://dx.doi.org/10.3837/tiis.2013.06.009

Copyright ⓒ 2013 KSII

1. Introduction

A chaotic map is a kind of dynamical system, the output of which is ergodic and sensitive to initial conditions [1]. The properties of a chaotic map make it useful for information encryption. Early in 1952, Shannon proposed mixing transformations based on the stretch-and-fold mechanism [2]. This method actually generates a chaotic system and achieves random permutations. The confusion and diffusion based on the chaotic map have strong chaos, which allows the achievement of high security for the cryptosystems [3, 4, 5]. The chaotic map also is used extensively in image encryption and steganography. The ergodicity of the chaotic map ensures that every input and every output correspond one-to-one, and the sensitivity to initial conditions provides high security for the cryptosystem. By scrambling and diffusing all the pixels in the plain image, the chaotic map generates a cipher image with strong chaos.

There are two kinds of methods in how to build the mapping function by a chaotic system in image encryption. The first kind transforms the coordinates of the pixels directly. That is to say, the coordinates of a pixel in the plain image are the input of the chaotic map, the mapping system outputs new coordinates, which indicate the location of the pixel in the cipher image. Since the coordinates of the pixel of gray image compose a two-dimensional vector, these encryption methods need two-dimensional chaotic maps. Fridrich proposed an image-encrypting scheme using the Baker map [6]. In this scheme, all the coordinates of the pixels in the plain image are transformed to different ones by the Baker map, generating a highly-chaotic cipher image. However, in order to enhance the security of the cryptosystem, Fridrich proposed that any two-dimensional chaotic map can be extended to three dimensions by modifying the pixels' gray values as well as permuting their locations. These scrambling methods are simple and fast, but the key space is usually low. Besides, chaotic systems have period, which decrease the security of the cryptosystem. In order to enhance the security, Zhu proposed to permute the pixels in bit-level [7]. In this method, each bit of the pixel is permuted by a two-dimensional chaotic map independently. This method improves the security of the cryptosystem, but the computational complexity is high.

Another kind of methods uses the chaotic map for encryption indirectly. In these methods, the input of the chaotic map is not the coordinates of the pixels, but initial values of the chaotic system. Using these initial values, the chaotic system outputs a series of numbers. The order of these numbers indicates the order of the pixels' locations in the cipher image. Since the chaotic system is highly sensitive to the initial states, a large key space is achieved. The standard map [8, 9], the logistic map [10, 11, 12], and the sine map [13, 14] can be used and achieve high security. But in these methods, the operations of the chaotic map are based on decimal numbers with high precision, and the iterations of the mapping algorithm are excessive. Therefore, the computational complexity is high.

In this paper, a chaotic map which has high security and low computational complexity is proposed. In the proposed scheme, the cat map is used. The map $X^{(n)} = AX^{(n-1)}$, in which

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \tag{1}$$

is defined as cat map [15]. The cat map is a kind of two-dimensional automorphism group. It has an ergodic hierarchy and strong chaos [16]. The cat map has been used extensively to

encrypt images [6, 17]. Pang proposed an image-encrypting scheme based on the cat map in the discrete wavelet transformation domain [18]. Chen extended the 2D cat map to a 3D cat map [19]. In this scheme, the image is piled up to three dimensions and is scrambled by the 3D cat map. This scheme achieved higher security and performs the computations faster. Chang proposed a copyright protection scheme for image based on the cat map. This method embeds the watermark image into the cover image by the cat map with tiny modifications to the cover image, thereby achieving a high-quality stego image [20].

Like many other chaotic maps, the cat map has periodicity, i.e., although several iterations from the original image generate a strongly-chaotic cipher image, a certain number of iterations can recover the original image completely. This number of iterations is the period of the cat map. Periodicity can be used in recovering the original image, but it also leads to the insecurity of the cryptosystem. Attackers can process iterations from the cipher image, and the original image will be recovered in certain iterations. Chen analyzed the periodic performance of the generalized cat map and proposed some approaches to deal with its periodicity when encrypting an image, but the period of the cat map was still relatively short [21].

In this paper, we propose a concatenated torus automorphisms mechanism that increases the period of the chaotic map significantly, resulting in higher security of the cryptosystem. The format of the rest of the paper is as follows. Section 2 gives the notations that are used in the paper and some preliminary information. In Section 3, we analyze the security of the cryptosystem based on the conventional torus automorphism, and Section 4 presents the proposed, novel, concatenated torus automorphisms mechanism and schemes for its application in image encryption. The security of the cryptosystem is analyzed and simulated in Section 5. Our conclusions, based on the performance of the proposed scheme in comparison to other schemes, are presented in Section 6.

2. Notations and Preliminaries

Before the novel concatenated torus automorphisms mechanism is proposed, a brief introduction and analysis of torus automorphism is presented in this section. The notations that are used are provided in **Table 1**.

Table 1. Notations

Notation	Definition
f^N	Torus automorphism with module N
R	Recurrence time of the torus automorphism
A	Scrambling matrix of the torus automorphism
$I^{(p)}$	Plain image
$I^{(c)}$	Cipher image
$I^{(d)}$	Decrypted image
k_e	Number of iterations in the encrypting process
k_d	Number of iterations in the decrypting process
$p_p(x^{(0)}, y^{(0)})$	Gray value of the pixel with coordinates $(x^{(0)}, y^{(0)})$ in the plain image
$p_c(x^{(k_e)}, y^{(k_e)})$	Gray value of the pixel with coordinates $(x^{(k_e)}, y^{(k_e)})$ in the cipher image
$p_d(x^{(k_d)}, y^{(k_d)})$	Gray value of the pixel with coordinates $(x^{(k_d)}, y^{(k_d)})$ in the decrypted image

Copyright ⓒ 2013 KSII

The two-dimensional automorphism f^1 of group G_1, denoted as $f^1: G_1 \to G_1,\ G_1 = [0,1)\times[0,1) \subset R^2$, is defined as a torus automorphism if the following is satisfied:

$$f^1: \begin{bmatrix} s_1' \\ s_2' \end{bmatrix} = A \cdot \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \bmod 1, \text{ where } A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \tag{2}$$

and $[s_1 \quad s_2]$ $[s_1' \quad s_2'] \in G_1$. The notation $x \bmod 1$ is the fractional part of the number x. In addition, $a_{ij} \in Z$ $(i, j = 1,2)$, $\det(A) = 1$, and the eigenvalues λ_1 and λ_2 of matrix A should satisfy $\lambda_{1,2} \notin \{-1,0,1\}$. In this paper, matrix A is defined as a scrambling matrix. The parameter $r = a_{11} + a_{22}$ is defined as the trace of matrix A. Percival and Vivaldi proved that the torus automorphism f^1 has strong chaos when $r^2 > 4$ [22].

Assuming that the initial state of the torus automorphism is $s^{(0)} = [s_1^{(0)} \quad s_2^{(0)}]$, a series of iterations from $s^{(0)}$ form a dynamical system, as shown in the following:

$$f^1: s^{(n)} = As^{(n-1)} = A^n s^{(0)} \Rightarrow \begin{bmatrix} s_1^{(n)} \\ s_2^{(n)} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^n \cdot \begin{bmatrix} s_1^{(0)} \\ s_2^{(0)} \end{bmatrix} \bmod 1, \tag{3}$$

where n is a positive integer. State $s^{(n)}$ is the n^{th} state from the initial state $s^{(0)}$, and $s^{(n)} = \begin{bmatrix} s_1^{(n)} & s_2^{(n)} \end{bmatrix}$. When n varies from 1 to infinity, the dynamical system forms a set of orbits $\{s^{(0)}, s^{(1)}, \cdots\}$.

Since $s_1, s_2 \in [0,1)$, we can denote s as the following form:

$$s = [s_1 \quad s_2] = \begin{bmatrix} \frac{p_1}{q_1} & \frac{p_2}{q_2} \end{bmatrix} = \begin{bmatrix} \frac{x}{N} & \frac{y}{N} \end{bmatrix} \in G_1, \tag{4}$$

where p_i and q_i $(i = 1,2)$ are co-prime integers, and N is the least common multiple of q_1 and q_2. Therefore, (3) can be transformed into another form, i.e., $f^N: G_N \to G_N$, where $G_N = \{0,1,\cdots,N-1\} \times \{0,1,\cdots,N-1\}$. The torus automorphism f^N is shown as the following:

$$f^N: \ X^{(n)} = A \cdot X^{(n-1)} = A^n \cdot X^{(0)} \Rightarrow \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^n \cdot \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} \bmod N, \tag{5}$$

where $X^{(0)} = \begin{bmatrix} x^{(0)} & y^{(0)} \end{bmatrix}$ is the initial state, $X^{(n)} = \begin{bmatrix} x^{(n)} & y^{(n)} \end{bmatrix}$ is the n^{th} state, and $X^{(0)}, X^{(n)} \in G_N$.

The one-module torus automorphism f^1, shown as (3), could be periodic. That is to say, for the set $\{s^{(0)}, s^{(1)}, \cdots\}$, an integer R may exist such that $s^{(0)} = s^{(R)}$. The necessary and sufficient condition for the torus automorphism f^1 to be periodic is that the initial state $s^{(0)}$ be composed of a pair of rational numbers. If the period of torus automorphism f^1 is R, then the

period of f^N is also R, under the condition that the scrambling matrices A in (3) and (5) are equal. Also, the period R is defined as the recurrence time of the torus automorphism. The recurrence time of the torus automorphism is usually found by simulations.

Meanwhile, since the scrambling matrix A in (5) is restricted by the conditions $\det(A)=1$ and its trace, the two-dimensional torus automorphism, f^N, is actually a two-parameter map. Therefore, (5) can be generalized as the following form [19]:

$$f^N:\ X^{(n)}=A\cdot X^{(n-1)}=A^n\cdot X^{(0)} \Rightarrow \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix}=\begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix}^n \cdot \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} \mod N. \tag{6}$$

3. Security Analysis of the Cryptosystem Based on the Torus Automorphism

Since the torus automorphism f^N is periodic with a period of R, R iterations from the initial state $X^{(0)}$ will return back to itself, i.e., $X^{(R)}=A^R X^{(0)}=X^{(0)}$. The period R of the torus automorphism f^N depends on the values of the scrambling matrix A and the module N. Now, we give the following theorem related to the period of the torus automorphism and its proof.

THEOREM 1: Two N-module torus automorphisms, f_α^N and f_β^N, are completely the same if the values of a and b in their scrambling matrices are correspondingly equal (mod N).

PROOF

Assume that there are two scrambling matrices, of which elements are correspondingly equal (mod N), as shown in the following:

$$A_\alpha=\begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \text{ and } A_\beta=\begin{bmatrix} 1 & a+kN \\ b+kN & (a+kN)(b+kN)+1 \end{bmatrix},$$

where $a,b,k\in Z$, and N is the module of the torus automorphisms. Then, the two torus automorphisms, f_α^N and f_β^N, with scrambling matrices of A_α and A_β, respectively, can be denoted as:

$$f_\alpha^N:\ \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix}=A_1\begin{bmatrix} x^{(n-1)} \\ y^{(n-1)} \end{bmatrix}=\begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix}\begin{bmatrix} x^{(n-1)} \\ y^{(n-1)} \end{bmatrix}=\begin{bmatrix} x^{(n-1)}+ay^{(n-1)} \\ bx^{(n-1)}+aby^{(n-1)}+y^{(n-1)} \end{bmatrix} \mod N,$$

$$\text{and } f_\beta^N:\ \begin{bmatrix} x^{(n)} \\ y^{(n)} \end{bmatrix}=A_2\begin{bmatrix} x^{(n-1)} \\ y^{(n-1)} \end{bmatrix}=\begin{bmatrix} 1 & a+kN \\ b+kN & (a+kN)(b+kN)+1 \end{bmatrix}\begin{bmatrix} x^{(n-1)} \\ y^{(n-1)} \end{bmatrix}$$

$$=\begin{bmatrix} x^{(n-1)}+ay^{(n-1)}+kNy^{(n-1)} \\ bx^{(n-1)}+kNx^{(n-1)}+aby^{(n-1)}+akNy^{(n-1)}+bkNy^{(n-1)}+k^2N^2y^{(n-1)}+y^{(n-1)} \end{bmatrix}$$

$$=\begin{bmatrix} x^{(n-1)}+ay^{(n-1)} \\ bx^{(n-1)}+aby^{(n-1)}+y^{(n-1)} \end{bmatrix} \mod N.$$

It can be seen that f_α^N is exactly the same as f_β^N. Therefore, Theorem 1 is proven.

Copyright ⓒ 2013 KSII

For example, consider that the modules of the two torus automorphisms, f_{α}^{16} and f_{β}^{16}, are both 16, and their scrambling matrices are:

$$A_{\alpha}=\begin{bmatrix}1 & 5\\ 7 & 36\end{bmatrix} \text{ and } A_{\beta}=\begin{bmatrix}1 & 21\\ 23 & 484\end{bmatrix}, \tag{7}$$

respectively. Note that the elements, a and b, in A_{α} and A_{β} are correspondingly equal (mod 16). Assuming that $X^{(0)}=\begin{bmatrix}7 & 11\end{bmatrix}$ is an initial state chosen arbitrarily, the states of the two torus automorphisms with the same initial state $X^{(0)}$ are shown in **Table 2**. We can see that the outputs of the two torus automorphisms are exactly the same, and the recurrence time is 12 for both of them.

Table 2. Examples of Two Torus Automorphisms

	$X^{(0)}$	$X^{(1)}$	$X^{(2)}$	$X^{(3)}$	$X^{(4)}$	$X^{(5)}$	$X^{(6)}$	$X^{(7)}$	$X^{(8)}$	$X^{(9)}$	$X^{(10)}$	$X^{(11)}$	$X^{(12)}$
f_{α}^{16}	7	14	15	13	2	13	15	14	7	5	2	5	7
	11	13	6	1	15	10	3	5	6	9	7	10	11
f_{β}^{16}	7	14	15	13	2	13	15	14	7	5	2	5	7
	11	13	6	1	15	10	3	5	6	9	7	10	11

Torus automorphisms can be used to encrypt images. Assuming that the size of the original grayscale image, $I^{(\mathrm{p})}$, is $N\times N$, the encrypting process scrambles $I^{(\mathrm{p})}$ by torus automorphism and obtains a highly-chaotic cipher image $I^{(\mathrm{c})}$. Any pixel with coordinates $(x^{(0)},y^{(0)})$ in $I^{(\mathrm{p})}$ is transferred to the new location $(x^{(k_e)},y^{(k_e)})$ in $I^{(\mathrm{c})}$ by the following map:

$$f^{N}:\begin{bmatrix}x^{(k_e)}\\ y^{(k_e)}\end{bmatrix}=A^{k_e}\cdot\begin{bmatrix}x^{(0)}\\ y^{(0)}\end{bmatrix} \mod N \quad x^{(0)},y^{(0)},x^{(k_e)},y^{(k_e)}\in G_N, \tag{8}$$

where N is the size of the image, and k_e is the iteration number of the encrypting process. Assuming that the recurrence time of the torus automorphism is R, the decrypting process recovers the original image by processing $R-k_e$ iterations from the cipher image $I^{(\mathrm{c})}$. Therefore, the decrypting algorithm is:

$$\begin{bmatrix}x^{(k_d)}\\ y^{(k_d)}\end{bmatrix}=A^{k_d}\cdot\begin{bmatrix}x^{(k_e)}\\ y^{(k_e)}\end{bmatrix} \mod N \quad x^{(k_e)},y^{(k_e)},x^{(k_d)},y^{(k_d)}\in G_N, \tag{9}$$

where $k_d=R-k_e$, and $\begin{bmatrix}x^{(k_d)} & y^{(k_d)}\end{bmatrix}=\begin{bmatrix}x^{(0)} & y^{(0)}\end{bmatrix}$; therefore, the original image is recovered successfully. To achieve a successful decryption, the following two requirements must be satisfied: 1) the scrambling matrix, A, must be consistent in the encrypting and decrypting processes, and 2) the sum of the iterations in the encrypting process and in the decrypting process must be equal to R, i.e., $k_e+k_d=R$.

Therefore, both the scrambling matrix and the iteration number can act as secret keys in the cryptosystem. For the scrambling matrix, there are two independent parameters, i.e., a and b, as shown in (6). Since the same scrambling matrices (mod N) result in identical torus automorphism, the ranges of a and b are both N. For the iteration number k_d in the

decrypting process, since $k_d = R - k_e$, the range of k_d is $1 \le k_d \le R-1$. If both A and k_d are secret keys of the image encrypting scheme, the total size S of the key space is:

$$S = N^2 (R-1). \tag{10}$$

In the experiment below, the size of the original image is 128×128. The scrambling matrix is A_α, as shown in (7). The recurrence time, R, is 96 when $N = 128$. The experimental results are shown as **Fig. 1**. In this figure, (a) is the original image, and (b), (c), (d) are scrambled images that are iterated 1, 50, and 96 times from the original image, respectively. We can see that images (b) and (c) are chaotic images, while (d) is exactly the same as (a). This means that R iterations from the original image will recover the original image, while the intermediate images are chaotic.

Assuming that the cipher image is (c), i.e., the number of iterations in the encryption is $k_e = 50$, we use the following two matrices:

$$A_\chi = \begin{bmatrix} 1 & 133 \\ 135 & 17956 \end{bmatrix} \text{ and } A_\delta = \begin{bmatrix} 1 & 6 \\ 7 & 43 \end{bmatrix},$$

to decrypt image (c). The decrypted images are (e) and (f) in Fig. 1. We see that (e) is the same as (a), since the elements in A_χ are correspondingly equal to those in A_α (mod 128), while (f) is chaotic because a different scrambling matrix, A_δ, is used for decryption.

(a) (b) (c) (d) (e) (f)

Fig. 1. Experimental Results of Encryption and Decryption Using Torus Automorphism

4. Image Encryption Based on the Concatenated Torus Automorphisms

One of the advantages of image encryption using the torus automorphism is that the successful decryption can be achieved by processing the right number of iterations from the cipher image with the right scrambling matrix. But two weaknesses lead to the insecurity of the cryptosystem. The first weakness is that the scrambling matrices are equal periodically, i.e., the period is the module N, and the second weakness is that the recurrence time of the conventional torus automorphism is usually short. In this section, we propose a chaotic map based on the concatenated torus automorphisms. The novel scheme provides higher security for the cryptosystem.

4.1 Concatenated Torus Automorphisms Mechanism

In the following, a theorem related to the concatenated torus automorphisms is proposed.

THEOREM 2: Assuming that there are L torus automorphisms, f_1^N, f_2^N, …, f_L^N, the scrambling matrix A_i of f_i^N $(i = 1, 2, \cdots, L)$ is

Copyright ⓒ 2013 KSII

$$A_i = \begin{bmatrix} 1 & a_i \\ b_i & a_i b_i + 1 \end{bmatrix}, \tag{11}$$

where $a_i, b_i \in Z$, $A_1 \neq A_2 \neq \cdots \neq A_L$, and the recurrence time of f_i^N is R_i. For any initial state $\begin{bmatrix} x^{(0)} & y^{(0)} \end{bmatrix} \in G_N$, if it is iterated k_1 times by A_1, k_2 times by A_2, …, k_L times by A_L, and the result $\begin{bmatrix} x^{(k_e)} & y^{(k_e)} \end{bmatrix}$ is obtained, then the initial state, $\begin{bmatrix} x^{(0)} & y^{(0)} \end{bmatrix}$, can be recovered if $\begin{bmatrix} x^{(k_e)} & y^{(k_e)} \end{bmatrix}$ is iterated $R_L - k_L$ times by A_L, $R_{L-1} - k_{L-1}$ times by A_{L-1}, …, and $R_1 - k_1$ times by A_1.

PROOF

$$\begin{bmatrix} x^{(k_e)} \\ y^{(k_e)} \end{bmatrix} = A_L^{\,k_L} \cdot A_{L-1}^{\,k_{L-1}} \cdots\cdots A_1^{k_1} \cdot \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} \mod N.$$

$$A_1^{R_1-k_1} \cdot A_2^{R_2-k_2} \cdots\cdots A_L^{R_L-k_L} \cdot \begin{bmatrix} x^{(k_e)} \\ y^{(k_e)} \end{bmatrix} = A_1^{R_1-k_1} \cdot A_2^{R_2-k_2} \cdots\cdots A_L^{R_L-k_L} \cdot A_L^{\,k_L} \cdot A_{L-1}^{\,k_{L-1}} \cdots\cdots A_1^{k_1} \cdot \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix}$$

$$= A_1^{R_1} \cdot A_2^{R_2} \cdots\cdots A_L^{R_L} \cdot \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} = \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} \mod N.$$

From Theorem 2, we know that several different scrambling matrices consist of a new dynamical system that is still periodic. The period R_c of the concatenated torus automorphisms is:

$$R_c = R_1 + R_2 + \cdots + R_L. \tag{12}$$

It is clear that the period of the concatenated torus automorphisms has been increased significantly. In order to ensure that all of the intermediate images are highly chaotic, the following condition must be satisfied:

$$(1 \le k_1 < R_1) \bigcap (1 \le k_2 < R_2) \bigcap \cdots \bigcap (1 \le k_L < R_L). \tag{13}$$

4.2 Image Encryption Based on the Two-Dimensional Concatenated Torus Automorphisms

In this section, we use the proposed concatenated torus automorphisms to encrypt images. Assuming that the size of the plain image, $I^{(p)}$, is $N \times N$, in encrypting process, all the pixels in the plain image are scrambled by the proposed concatenated torus automorphisms, i.e., the values of the pixels' coordinates are iterated k_1 times by A_1, k_2 times by A_2, …, and k_L times by A_L, generating the cipher image that has strong chaos. When decryption is performed, the cipher image is scrambled $R_L - k_L$ times by A_L, $R_{L-1} - k_{L-1}$ times by A_{L-1}, …, and $R_1 - k_1$ times by A_1, after which the original image is recovered completely. The framework of the encrypting and decrypting processes of the proposed scheme is shown as **Fig. 2**.

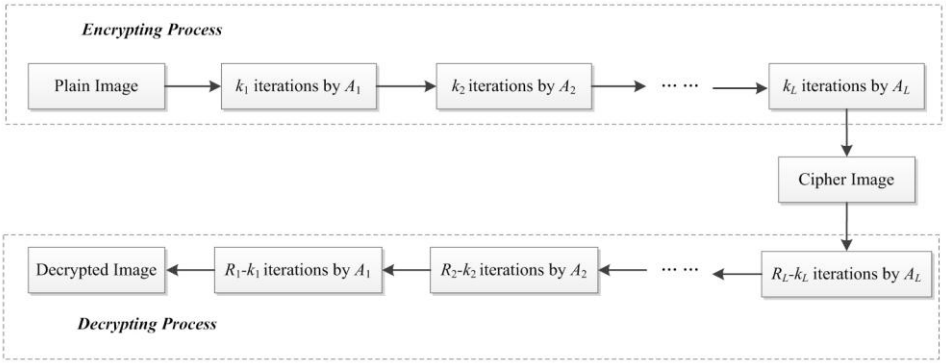


Fig. 2. Framework of the Cryptosystem Based on the 2D Concatenated Torus Automorphisms

Therefore, any pixel with the coordinates $(x^{(0)}, y^{(0)})$ in the plain image will be transferred to $(x^{(k_e)}, y^{(k_e)})$ in the cipher image by the following encrypting algorithm:

$$\begin{bmatrix} x^{(k_e)} \\ y^{(k_e)} \end{bmatrix} = A_L^{k_{e_L}} A_{L-1}^{k_{e_{L-1}}} \cdots A_1^{k_{e_1}} \begin{bmatrix} x^{(0)} \\ y^{(0)} \end{bmatrix} \mod N, \quad x^{(0)}, y^{(0)}, x^{(k_e)}, y^{(k_e)} \in G_N . \tag{14}$$

The total number of the iterations in the encrypting process is $k_e = k_{e_1} + k_{e_2} + \cdots k_{e_L}$.

The decrypting algorithm is:

$$\begin{bmatrix} x^{(k_d)} \\ y^{(k_d)} \end{bmatrix} = A_1^{k_{d_1}} A_2^{k_{d_2}} \cdots A_L^{k_{d_L}} \begin{bmatrix} x^{(k_e)} \\ y^{(k_e)} \end{bmatrix} \mod N, \quad x^{(k_e)}, y^{(k_e)}, x^{(k_d)}, y^{(k_d)} \in G_N , \tag{15}$$

where $k_{d_1} = R_1 - k_{e_1}$, $k_{d_2} = R_2 - k_{e_2}$, …, $k_{d_L} = R_L - k_{e_L}$. The total number of iterations in the decrypting process is $k_d = k_{d_1} + k_{d_2} + \cdots + k_{d_L}$. It is clear that $k_e + k_d = R_1 + R_2 + \cdots + R_L = R_c$. From Theorem 2, we know that $(x^{(k_d)}, y^{(k_d)}) = (x^{(0)}, y^{(0)})$.

An example is shown below. The concatenated torus automorphisms are composed of three torus automorphisms, which are f_1^N, f_2^N, and f_3^N. Their scrambling matrices are shown as the following:

$$A_1 = \begin{bmatrix} 1 & 9 \\ 12 & 109 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 7 \\ 5 & 36 \end{bmatrix}, A_3 = \begin{bmatrix} 1 & 4 \\ 6 & 25 \end{bmatrix}. \tag{16}$$

When the module is 128 (the image size), the recurrence times of the three torus automorphisms are $R_1 = 128$, $R_2 = 96$, and $R_3 = 64$, respectively. In the encrypting process, the original image is scrambled $k_{e_1} = 40$ times by f_1^N, $k_{e_2} = 8$ times by f_2^N, and $k_{e_3} = 13$ times by f_3^N. **Fig. 3** shows the experimental results. In this figure, (a) is the original image, (b) and (c) are the scrambled images with one and 40 iterations by A_1 from (a), respectively, (d) is the scrambled images with eight iterations by A_2 from (c), and (e) is the scrambled images with 13 iterations by A_3 from (d). Image (e) is the cipher image, which has been iterated 61 times from the original image by the three scrambling matrices, and it is highly chaotic. In the decrypting process, (e) is iterated $k_{d_3} = R_3 - k_{e_3} = 51$ times by A_3, thereby obtaining image (f);

Copyright ⓒ 2013 KSII

(f) is iterated $k_{d_2} = R_2 - k_{e_2} = 88$ times by A_2, thereby obtaining (g); and (g) is iterated $k_{d_1} = R_1 - k_{e_1} = 88$ times by A_1, thereby obtaining (h). Image (h) is the decrypted image, and it is evident that it is exactly the same as (a). From (a) to (h), there was a total of 288 iterations, which is exactly the sum of R_1, R_2, and R_3. Only image (h) is exactly the same as (a), while all the intermediate images are strongly chaotic.

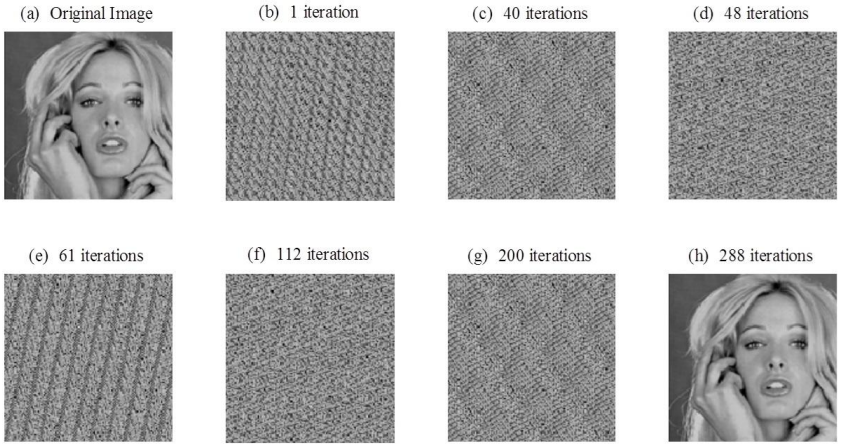


Fig. 3. Experimental Results of the Image Encryption Using the 2D Concatenated Torus Automorphisms

4.3 Image Encryption Based on the Three-Dimensional Concatenated Torus Automorphisms

The cryptosystem proposed above scrambles all the pixels in the plain image but doesn't change their gray values, which leads to the same histogram distribution as that of the plain image, i.e., the cipher image using a 2D chaotic map has a non-uniform histogram. In this section, an improved scheme that uses the 3D concatenated torus automorphisms is proposed. In this scheme, both the positions and the gray values of the pixels in the plain image are modified by the concatenated torus automorphisms. The framework of the 3D concatenated chaotic map is shown as **Fig. 4**.

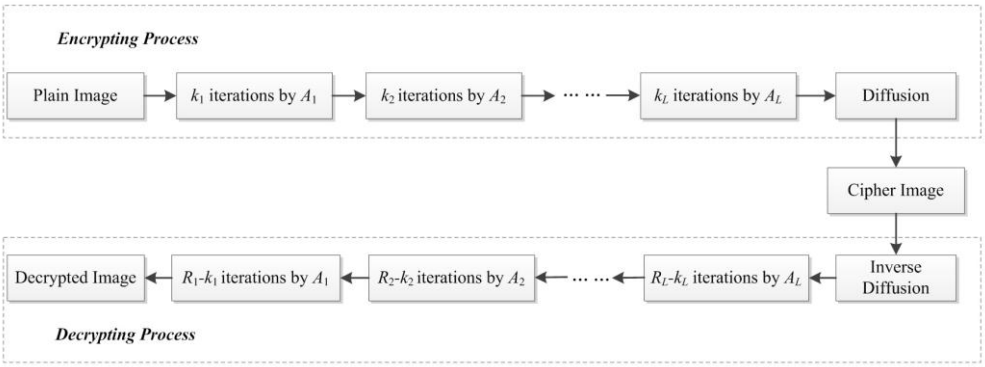


Fig. 4. Framework of the Cryptosystem Based on the 3D Concatenated Torus Automorphisms

When encrypting, the pixel with coordinate $(x^{(0)}, y^{(0)})$ in the plain image is permuted to $(x^{(k_e)}, y^{(k_e)})$ in the cipher image by (14). Meanwhile, the gray value of the pixel is modified by the diffusion function F. In order to provide high security and to achieve successful decryption, the diffusion function F must satisfy the following requirements:

- The modifications to pixels' gray values by function F must provide a uniform

histogram distribution to the cipher image.

- Function F must have the characteristic of diffusion.
- The inverse operation of the diffusion function F must recover the gray values of the plain image.

Therefore, the output of the diffusion function F in our proposed scheme depends on the pixel's gray value in the plain image, the gray value of the previous pixel in the cipher image, and the coordinates of the pixel in the plain image and cipher image, shown as the following:

$$F: p_c(l_{cipher}) = p_p(l_{plain}) + \lambda p_c(l_{cipher} - 1) + \left(x^{(k_e)} - x^{(0)}\right) + \left(y^{(k_e)} - y^{(0)}\right) \mod 256 , \quad (17)$$

where $l_{cipher} = N(x^{(k_e)} - 1) + y^{(k_e)}$, and $l_{plain} = N(x^{(0)} - 1) + y^{(0)}$. In (17), $p_c(l_{cipher})$ is the gray value of the $l_{cipher}{}^{th}$ pixel in the cipher image, $p_p(l_{plain})$ is the gray value of the $l_{plain}{}^{th}$ pixel in the plain image. The parameter λ is an integer, which expands the effect of the previous pixel to the current pixel in the cipher image. In order to compute the value of the first pixel in the cipher image, of which $l_{cipher} = 1$, the value of $p_c(0)$ should be pre-determined. Then, the modification to the pixel's gray value, implemented by (17), can be diffused to the whole image when l_{cipher} varies from 1 to N^2. The diffusion process can be implemented several rounds. To start the second round, we take the diffused image as plain image. The initial value $p_c(0)$ of the second round is the gray value of the last pixel of the diffused image of the previous round. Then, we can implement the second round by (17). This process can be implemented several rounds.

In decryption process, the pixels' locations are recovered by (15), and their gray values are recovered by the inverse operation of function F, which can be shown as the following:

$$F^{-1}: p_d(l_{decrypted}) = p_c(l_{cipher}) - \lambda p_c(l_{cipher} - 1) - \left(x^{(k_e)} - x^{(k_d)}\right) - \left(y^{(k_e)} - y^{(k_d)}\right) \mod 256 , \quad (18)$$

where $l_{cipher} = N(x^{(k_e)} - 1) + y^{(k_e)}$, and $l_{decrypted} = N(x^{(k_d)} - 1) + y^{(k_d)}$.

Fig. 5 shows the experimental results. The concatenated torus automorphisms are the same as those in the example in Section 4.2, and the diffusion algorithm is shown as (17), where λ=7. We can see that only one iteration of the concatenated chaotic map generates a highly-chaotic image, shown as image (b). The cipher image, which is iterated 61 times and is shown as (c), is highly chaotic. The decrypted image, which is obtained by (15) and (18), is the same as the plain image. **Fig. 6** shows the histograms of the images in **Fig. 5**, respectively. We can see that the cipher images have uniform histogram distributions.

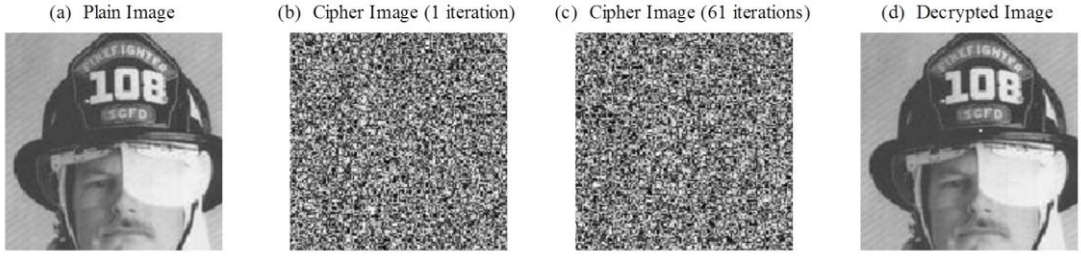


Fig. 5. Experimental Results of Image Encryption Using the 3D Concatenated Torus Automorphisms

Copyright ⓒ 2013 KSII

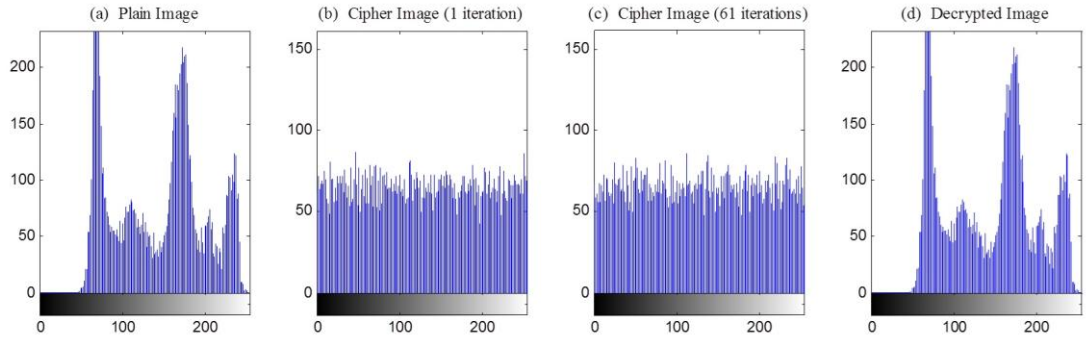


Fig. 6. Histograms of the Cipher Images Encrypted by the 3D Concatenated Torus Automrophisms

5. Analysis of the Statistical Performance and Computational Complexity

5.1 Statistical Performances

In this section, the security of the proposed schemes is analyzed statistically. The cryptosystems based on the proposed 2D and 3D concatenated torus automorphisms are both studied, and the examples presented in Sections 4.2 and 4.3 are used for simulation. For comparison, the following four schemes were analyzed:

a. 2D cat map [15]. This method inputs the coordinates of the pixels in the plain image to the mapping function and computes the coordinates of the pixels in the cipher image directly.

b. 3D cat map [17]. This method transforms the coordiantes of the pixels by the mapping function directly and modifies the pixels' gray values by performing an exclusive OR operation with random numbers.

c. 3D logistic map [12]. This method permutes the pixels in the plain image according to the order of a series of random numbers generated by the logistic map. Meanwhile, a diffusion operation is implemented to the pixels' gray values.

d. 3D bit-level permutation [7]. This method takes the same bits of all the pixels in the plain image as a group and permutes them by a cat map. Different bit groups are permuted by different cat maps. Meanwhile, the pixels' gray values of the permutated image are diffused by a logistic map.

5.1.1 Mean Square Error

The cipher image should look totally different from the plain image, and this difference can be measured by the mean square error (MSE) between the two images.

$$MSE = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[p_p(i,j) - p_c(i,j)\right]^2 , \tag{19}$$

where N is the size of the image, $p_p(i,j)$ and $p_c(i,j)$ are the gray values of the pixels with coordinates (i,j) in the plain image and the cipher image, respectively. It is clear that $MSE = 0$ when the plain image and the cipher image are the same. The greater the difference between them is, the larger the MSE value is.

The *MSE* values between the plain images and their cipher images are shown in **Table 3**. We can see that the proposed 2D concatenated torus automorphisms provide the same effect as the 2D cat map [15], and the proposed 3D concatenated torus automorphisms have almost the same effect as the schemes in [17], [12], and [7].

Therefore, from these data, we can see that the cipher images encrypted by our novel concatenated chaotic maps are totally different from the plain images, and the encrypting effects are as good as the existing methods.

Table 3. Mean Square Error of the Cipher Images

Image	2D Cat Map [15]	3D Cat Map [17]	3D Logistic Map [12]	3D Bit-Level Permutation [7]	Proposed (2D)	Proposed (3D)
Tiffany	3.38e+03	6.93e+3	7.38e+03	7.08e+03	3.37e+03	7.11e+03
Baboon	2.59e+03	6.70e+3	6.79e+03	6.72e+03	2.57e+03	6.77e+03
Parrot	4.21e+03	7.58e+3	7.60e+03	7.51e+03	4.20e+03	7.60e+03
Goldhill	4.71e+03	7.24e+3	8.15e+03	8.03e+03	4.71e+03	8.11e+03
Barbara	3.99e+03	7.23e+3	7.70e+03	7.81e+03	3.97e+03	7.79e+03
Peppers	6.45e+03	8.76e+3	9.14e+03	9.39e+03	6.41e+03	9.25e+03
Cameraman	7.34e+03	9.38e+3	9.24e+03	9.34e+03	7.34e+03	9.26e+03
Lena	5.32e+03	7.36e+3	8.93e+03	9.09e+03	5.37e+03	9.08e+03
Firefighter	6.38e+03	8.54e+3	8.90e+03	8.88e+03	6.33e+03	8.94+03

5.2.2 Histogram Distribution

The histograms of the plain images are usually nonuniform. In order to provide high security, the histograms of the cipher images should be uniform, which can be achieved by 3D chaotic maps. We use the standard deviation SD_{his} of the histogram distribution to measure its uniformity:

$$SD_{\text{his}} = \sqrt{\frac{1}{Z}\sum_{i=1}^{Z}\left[x_i - E(x)\right]}\ ,$$
$$E(x) = \frac{1}{Z}\sum_{i=1}^{Z} x_i\ , \tag{20}$$

where Z is the range of the gray value, and x_i ($i = 1,2,\cdots,Z$) is the amplitude of the histogram. **Table 4** shows the standard deviation of the histogram distribution of the plain images and the cipher images, respectively. We can see that the 2D cat map [15] and the proposed 2D concatenated chaotic map provide exactly the same histogram distributions to the cipher images as those of the plain images. This is because that they don't modify the pixels' gray values. The schemes proposed in [12], [7], and our 3D concatenated chaotic map have uniform histogram distributions, and the uniformity is better than that of the scheme in [17].

Table 4. Standard Deviation of the Histogram Distribution

Image	Plain Image	2D Cat Map [15]	3D Cat Map [17]	3D Logistic Map [12]	3D Bit-Level Permutation [7]	Proposed (2D)	Proposed (3D)
Tiffany	66.24	66.24	18.84	8.10	8.01	66.24	8.11
Baboon	65.00	65.00	8.42	8.61	7.80	65.00	7.88
Parrot	70.29	70.29	14.02	7.82	7.91	70.29	7.88
Goldhill	53.88	53.88	26.70	7.84	8.05	53.88	7.45
Barbara	52.49	52.49	16.12	8.06	8.22	52.49	7.94
Peppers	46.58	46.58	12.55	8.14	8.50	46.58	7.81
Cameraman	75.56	75.56	12.08	7.91	7.35	75.56	7.95

Copyright ⓒ 2013 KSII

Lena	43.50	43.50	28.32	8.05	8.02	43.50	8.15
Firefighter	67.77	67.77	13.37	8.64	8,11	67.77	8.24

5.2.3 Correlation of the Adjacent Pixels

The adjacent pixels in a plain image are usually highly correlated, while the adjacent pixels in a scrambled image have a low correlation. **Fig. 7** shows the distributions of the adjacent pixels in the plain image and the cipher image, respectively, in which the cipher image is encrypted by the proposed 3D concatenated torus automorphisms. The adjacent pixels are taken in horizontal, vertical, and diagonal directions, respectively. We can see that the adjacent pixels in the plain image have high correlations, while those in the cipher image have low correlations.

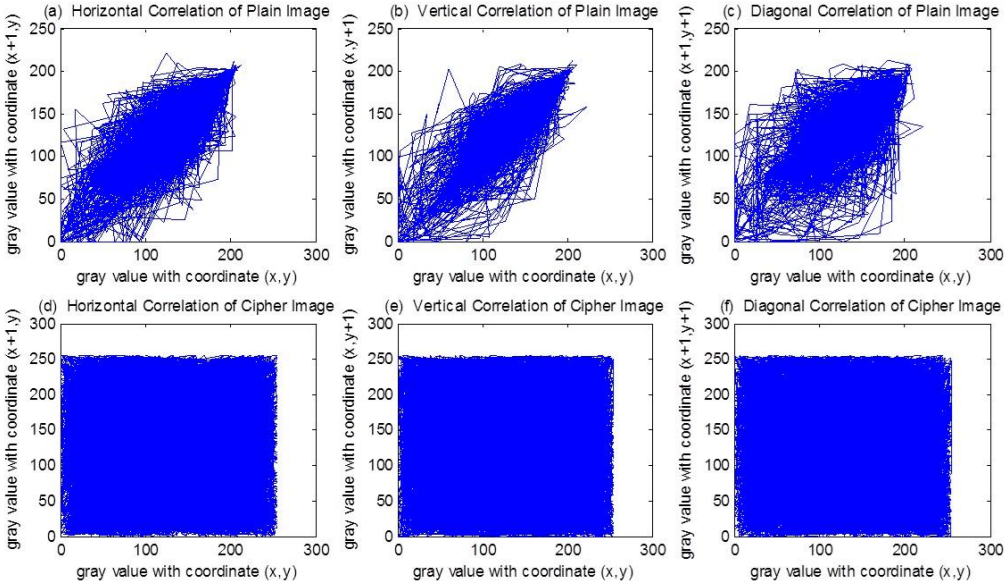


Fig. 7. Distributions of the Adjacent Pixels

In order to measure the correlation performances between adjacent pixels, the correlation coefficient r is computed by the following formula:

$$
\begin{aligned}
r &= \frac{\operatorname{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \\
\operatorname{cov}(x,y) &= \frac{2}{N^2}\sum_{i=1}^{N^2/2}\left[x_i - E(x)\right]\left[y_i - E(y)\right], \\
D(x) &= \frac{2}{N^2}\sum_{i=1}^{N^2/2}\left[x_i - E(x)\right], \\
E(x) &= \frac{2}{N^2}\sum_{i=1}^{N^2/2} x_i,
\end{aligned}
\tag{21}
$$

where x and y are the gray values of the adjacent pixels. Note that the adjacent pixels can be taken in horizontal, vertical, or diagonal directions. The smaller the correlation coefficient is,

the more chaotic the image is. **Table 5** shows the correlation coefficients of the six schemes. We can see that the plain images have high correlations between adjacent pixels, while all the cipher images encrypted by the six schemes have low correlations. This means that the proposed 2D and 3D concatenated chaotic maps provide the cipher images as strong chaos as the existing methods.

Table 5. Correlation Coefficients

	Image	Tiffany	Baboon	Parrot	Goldhill	Barbara	Peppers	Cameraman	Lena	Firefighter
horizontal	Plain Image	0.892	0.849	0.953	0.941	0.901	0.909	0.905	0.879	0.924
	2D Cat Map [15]	2.1e-2	2.1e-2	4.7e-3	-1.7e-3	9.2e-3	2.3e-2	6.1e-5	1.5e-2	2.8e-3
	3D Cat Map [17]	1.0e-2	1.0e-2	1.6e-3	-8.9e-4	6.9e-3	1.4e-2	-2.7e-4	6.9e-3	2.0e-3
	3D Logistic Map [12]	-1.7e-2	7.2e-3	-2.2e-4	-2.3e-2	1.6e-2	1.9e-2	3.0e-3	-8.0e-3	9.4e-3
	3D Bit-Level Permutation [7]	4.1e-3	4.7e-3	1.1e-2	-1.1e-2	5.8e-3	-7.5e-4	-3.1e-3	-2.5e-3	5.8e-3
	Proposed (2D)	-6.1e-3	-1.3e-2	4.6e-4	-1.1e-2	-4.3e-3	-4.4e-3	-7.2e-3	-3.9e-3	-2.6e-3
	Proposed (3D)	1.3e-2	-9.1e-3	2.5e-3	7.2e-3	1.6e-2	-1.7e-3	3.7e-3	-7.4e-3	8.4e-4
vertical	Plain Image	0.922	0.853	0.975	0.945	0.94	0.936	0.938	0.941	0.939
	2D Cat Map [15]	2.0e-3	2.0e-2	9.5e-3	5.4e-3	1.7e-2	1.7e-2	3.0e-3	1.6e-2	6.6e-3
	3D Cat Map [17]	3.2e-4	1.4e-2	4.7e-3	5.5e-3	1.1e-2	8.8e-3	2.7e-3	1.2e-2	4.9e-3
	3D Logistic Map [12]	-2.2e-3	-1.8e-3	-4.0e-4	-6.5e-3	1.1e-2	-3.2e-2	3.4e-3	-6.1e-3	-8.6e-3
	3D Bit-Level Permutation [7]	-2.3e-2	-1.2e-2	-1.6e-2	7.1e-3	7.9e-3	-7.8e-4	5.6e-3	-7.1e-3	9.5e-3
	Proposed (2D)	-2.3e-3	-2.1e-3	-5.3e-3	-2.6e-3	6.4e-4	-2.8e-3	-3.7e-3	-4.2e-3	-2.3e-3
	Proposed (3D)	1.6e-2	-1.9e-2	1.5e-2	3.5e-3	4.7e-4	1.0e-2	1.2e-2	1.0e-2	6.1e-3
diagonal	Plain Image	0.841	0.79	0.93	0.899	0.847	0.86	0.869	0.842	0.891
	2D Cat Map [15]	3.0e-3	2.1e-2	9.7e-3	5.8e-3	1.8e-2	1.8e-2	3.8e-3	1.8e-2	7.6e-3
	3D Cat Map [17]	2.9e-3	1.4e-2	5.0e-3	6.2e-3	1.4e-2	9.1e-3	5.1e-3	1.3e-2	5.4e-3
	3D Logistic Map [12]	-2.8e-2	3.6e-3	9.4e-3	-8.6e-3	4.4e-3	-1.2e-2	-1.1e-2	-1.1e-2	-1.3e-2
	3D Bit-Level Permutation [7]	-1.1e-2	-7.5e-3	-9.7e-3	-5.7e-4	-1.6e-2	-5.4e-3	-6.5e-3	7.6e-3	-5.0e-3
	Proposed (2D)	2.8e-3	-1.7e-3	-2.0e-3	-2.1e-3	-3.6e-3	-5.0e-3	-6.2e-3	1.4e-3	7.7e-4
	Proposed (3D)	2.9e-2	-7.7e-4	-4.4e-3	8.0e-3	-4.3e-3	-7.0e-4	-5.5e-4	-1.9e-2	-2.6e-3

5.2 Computational Complexity

The computational complexity of our proposed scheme is low. Most cryptosystems need high computing precision, because it provides large key space for the cryptosystems, such as the 3D Logistic Map [12] and the 3D Bit-Level Permutation [7]. But high computing precision leads to high computational complexity. For the proposed scheme, all the operations are integer operations modulo N, therefore, the computational complexity of the proposed scheme is lower than those of the existing methods.

The number of permutations of the proposed scheme is small. The 3D logistic map [12] permutes the pixels of the plain image twice (horizontally and vertically), and the 3D bit-level permutation permutes the plain image in bit level. Therefore, their computing times are longer than the proposed scheme. The computing time of our scheme is almost the same as that of the 3D cat map [17]. The computing time of Matlab simulations of the four encrypting schemes is shown in **Table 6**.

Copyright ⓒ 2013 KSII

Table 6. *Computing Time (s)*

Round	3D Cat Map [17]	3D Logistic Map [12]	3D Bit-Level Permutation [7]	Proposed (3D)
1	0.015350	0.023624	8.993344	0.016413
2	0.025744	0.033526	17.823442	0.027742
3	0.036862	0.042199	26.596119	0.037030
4	0.047480	0.052959	35.290206	0.046635
5	0.060780	0.060658	44.246981	0.056610

6. Security Analysis

In this section, the performances of the proposed scheme to resist attacks are analyzed.

6.1 Known-Plaintext Attack

The known-plaintext attack is an attack method where the attacker has both the plaintext and its ciphertext. Using this information, the attacker finds the secret keys.

In order to resist the known-plaintext attack, a cryptosystem should be sensitive to secret keys, and the key space should be large enough to resist brute-force attacks. In the following, we analyze the key performances of the proposed concatenated chaotic maps. As we said, both the scrambling matrix A and the iteration times k_d for decryption act as secret keys in our cryptosystem. First, we analyze the effect of the scrambling matrix. For the proposed 2D chaotic map, assume that the encryption is processed by (14), and the scrambling matrices are shown as (16). For decryption, we assume that the right numbers of iterations, k_{d_3}, k_{d_2}, and k_{d_1}, are implemented, but the scrambling matrices for decryption are shown as the following:

$$A_1 = \begin{bmatrix} 1 & 9 \\ 12 & 109 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 8 \\ 5 & 41 \end{bmatrix}, A_3 = \begin{bmatrix} 1 & 4 \\ 6 & 25 \end{bmatrix}. \quad (22)$$

Compared with (16), we see that only a_2 is different, and all the other parameters are the same. The experimental results are shown in **Fig. 8**. In this figure, (a) is the original image, (b) is the cipher image using scrambling matrices (16), and (c) and (d) are decrypted images using (16) and (22), respectively. We see that the same scrambling matrices in encryption and decryption lead to successful decryption, while a little difference of the scrambling matrices in the decrypting process leads to a strongly chaotic decrypted image, shown as (d). This means that the proposed cryptosystem is sensitive to the scrambling matrix. This phenomenon can be explained easily based on the sensitivity of the torus automorphism to the scrambling matrix [22]. The same situation exists for the proposed 3D concatenated chaotic map, since the permutations for the pixels in encryption and decryption are the same.

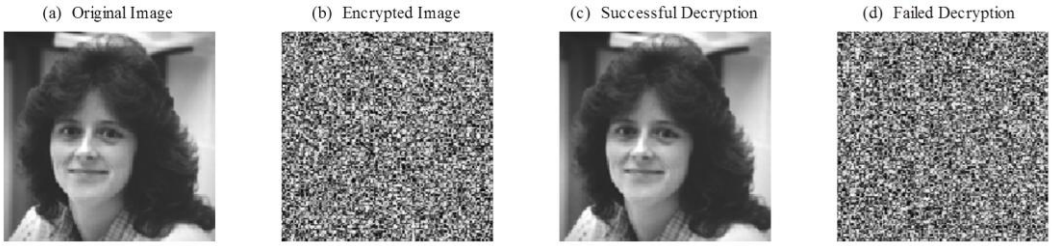


Fig. 8. Decryption Results by Different Scrambling Matrices

On another hand, incorrect iteration times also lead to failed decryption, since only the correct iteration times, which are the recurrence times of the concatenated torus automorphisms, can recover the original image, while all the intermediate images are strongly chaotic, as shown in **Fig. 3** and **Fig. 5** and **Tables 3-5**.

For a concatenated chaotic map with L torus automorphisms and module N, since the range of a_i and b_i are both N, the key space given by the scrambling matrices is N^{2L}. On the other hand, for the i^{th} torus automorphism of the concatenated chaotic map in the decrypting process, the correct number of iterations is $k_{d_i} = R_i - k_{e_i}$ ($i = 1,2,\cdots,L$). Since $1 \le k_{e_i} \le R_i - 1$, the range of k_{d_i} is $1 \le k_{d_i} \le R_i - 1$. Therefore, the key space given by the iteration times is $\prod_{i=1}^{L}(R_i - 1)$. Then, the total key spaces S_{con} for both our proposed 2D and 3D concatenated chaotic maps are:

$$S_{con} = N^{2L}\prod_{i=1}^{L}(R_i - 1). \tag{23}$$

By comparing (23) with (10), we can see that the key space of the proposed scheme is exactly the same as that of the conventional torus automorphism when the number L is one. The number of torus automorphisms in the proposed scheme depends on the security requirements, and the number can be as large as necessary. Assuming that $L = 5$, $N = 512$, and $R_i = 513$ ($i = 1,2,\cdots,5$), the key space is $S_{con} = 512^{15} = 2^{135}$. Therefore, the concatenated chaotic maps are secure enough to resist brute-force attacks.

6.2 Chosen-Plaintext Attack

Differential attack is one of the most popular chosen-plaintext attacks, which is often used to attack the image encrypting system. Assuming that there are two images with only one-bit difference, the two images are encrypted by the cryptosystem and two cipher images, C_1 and C_2, are obtained, respectively. By comparing the differences between C_1 and C_2, the attackers may trace the relationship between the plain image and the cipher image. Therefore, the more different the two cipher images are, the securer the cryptosystem is. Usually, we use the number of pixels change rate (*NPCR*) and the unified average changing intensity (*UACI*) to measure the differences between C_1 and C_2. The *NPCR* is defined as:

$$NPCR = \frac{\sum_{i,j=1}^{N} D(i,j)}{N^2}. \tag{24}$$

Copyright ⓒ 2013 KSII

In (24), $D(i, j)$ is defined as

$$D(i,j)=\begin{cases}1, & c_1(i,j)\neq c_1(i,j)\\ 0, & c_1(i,j)=c_1(i,j)\end{cases},$$

where $c_1(i, j)$ and $c_2(i, j)$ are the gray values with coordinates (i, j) in the cipher images C_1 and C_2, respectively.

The *UACI* is defined as:

$$UACI=\frac{\sum_{i,j=1}^{N}\left|c_1(i,j)-c_2(i,j)\right|}{255N^2} \tag{25}$$

In the following experiments, there are two plain images that are almost the same, with only one difference in the gray values of the last pixels, i.e., $p_1(N,N)=p_2(N,N)+1 \pmod{256}$, where $p_1(N,N)$ and $p_2(N,N)$ are the gray values of the pixels with coordinates (N,N) in the two plain images, respectively. The *NPCR* and *UACI* of different encrypting schemes are shown in **Table 7**. In the experiments, the diffusions are implemented five rounds. From the table we see that the *NPCR* and *UACI* of the first sound are low, no matter which encrypting scheme is used. This is because that the different pixel is in the last location, which obstructs the diffusion of the difference. For the proposed 3D concatenated chaotic map, the difference is diffused to the entire image after the second round, and high *NPCR* and *UACI* are achieved thereby. The 3D Logistic Map [12] and the 3D Bit-Level Permutation [7] have the similar effects, but the diffusing speeds are lower than that of the proposed scheme. For the 3D Cat Map [17], since the modifications to the gray values are not diffused, its *NPCR* and *UACI* are very low.

Table 7. *NPCR* and *UACI*

Round	3D Cat Map [17]		3D Logistic Map [12]		3D Bit-Level Permutation [7]		Proposed (3D)	
	NPCR	*UACI*	*NPCR*	*UACI*	*NPCR*	*UACI*	*NPCR*	*UACI*
1	6.10e-5	2.39e-7	0.0184	4.45e-4	6.10e-5	2.39e-7	6.10e-5	2.39e-7
2	6.10e-5	2.39e-7	0.9998	0.2245	0.8120	0.2739	1	0.3314
3	6.10e-5	2.39e-7	1	0.3309	0.9961	0.3340	0.9961	0.3324
4	6.10e-5	2.39e-7	0.9922	0.3376	0.9959	0.3348	0.9961	0.3347
5	6.10e-5	2.39e-7	1	0.3235	0.9963	0.3348	0.9902	0.3353

7. Conclusions

A novel cryptosystem based on the concatenated torus automorphisms was proposed in this paper. The proposed scheme concatenated several torus automorphisms, which increased the recurrence time of the chaotic map, thereby improving the security of the cryptosystem. Based on the novel concatenated chaotic map, two application schemes in image encryption were proposed, i.e., 2D and 3D concatenated torus automorphisms. Analyses and simulations showed that the chaos and key space of the proposed schemes are as good as the existing methods, while the speed of diffusion of the proposed schemes is faster, and the computational complexity is lower, compared with the existing schemes.

References

[1] G. Jakimoski and L. Kocarev, "Chaos and cryptography: block encryption ciphers based on chaotic maps," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 2, pp. 163-169, 2001.

[2] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal (Bell System Tech.)*, vol. 28, no. 4, pp. 656-715, 1949.

[3] B. Jovic and C. P. Unsworth, "Fast synchronisation of chaotic maps for secure chaotic communications," *Electronics Letters*, vol. 46, no. 1, pp. 49-50, 2010.

[4] K. W. Wong, Q. Z. Lin and J. Y. Chen, "Simultaneous arithmetic coding and encryption using chaotic maps," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 2, pp. 146-150, 2010.

[5] G. A. Sathishkumar, S. Ramachandran and K. B. Bagan, "Image encryption using random pixel permutation by chaotic mapping," in *Proc. of IEEE Symposium on Computers & Informatics (ISCI)*, pp. 247-251, 2012.

[6] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," *International Journal of Bifurcation and Chaos*, vol. 8, no 6, pp. 1259-1284, 1998.

[7] Z. L. Zhu, W. Zhang, K. W. Wong and H. Yu, "A chaos-based symmetric image encryption scheme using a bit-level permutation," *Information Science*, vol. 181, no. 6, pp. 1171-1186, 2011.

[8] R. S. Ye and H. Q. Huang, "A novel image shuffling and watermarking scheme based on standard map," in *Proc. of International Conference on Information Engineering and Computer Science (ICIECS)*, pp. 1-4, 2009.

[9] V. Patidar, G. Purohit, K. K. Sud and N. K. Pareek, "Image encryption through a novel permutation-substitution scheme based on chaotic standard map," in *Proc. of International Workshop on Chaos-Fractals Theories and Applications (IWCFTA)*, pp. 164-169, 2010.

[10] Y. Sun and G. Y. Wang, "An image encryption scheme based on modified logistic map," in *Proc. of Fourth International Workshop on Chaos-Fractals Theories and Applications (IWCFTA)*, pp. 179-182, 2011.

[11] A. H. Zhu and L. Li, "Improving for chaotic image encryption algorithm based on logistic map," in *Proc. of International Conference on Environmental Science and Information Application Technology (ESIAT)*, vol. 3, pp. 211-214, 2010.

[12] X. Y, Wang, L. Teng and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Processing*, vol. 92, no. 4, pp. 1101-1108, 2012.

[13] Y. Y. Mao and X. Chen, "An encryption algorithm of chaos based on sine square mapping," in *Proc. of Fourth International Symposium on Computational Intelligence and Design (ISCID)*, pp. 131-134, 2011.

[14] D. E. Goumidi and F. Hachouf, "Modified confusion-diffusion based satellite image cipher using chaotic standard, logistic and sine maps," in *Proc. of 2nd European Workshop on Visual Information Processing (EUVIP)*, pp. 204-209, 2010.

[15] V. I. Arnold and A. Avez, *Ergodic Problems of Classical Mechanics*, Benjamin, New York, 1968.

[16] P. Akritas, I. E. Antoniou and G. P. Pronko, "On the torus automorphisms: analytic solution, computability and quantization," *Chaos, Solitons and Fractals*, vol. 12, no. 14, pp. 2805-2814, 2001.

[17] L. H. Zhu, W. Z. Li, L. J. Liao and H. Li, "A novel algorithm for scrambling digital image based on cat chaotic mapping," in *Proc. of International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pp. 601-604, 2006.

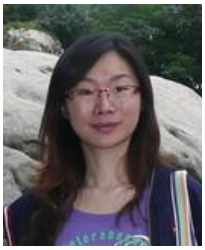
[18] C. J. Pang, "An image encryption algorithm based on discrete wavelet transform and two dimension cat mapping," in *Proc. of International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC)*, pp. 711-714, 2009.

[19] G. Chen, Y. Mao and C. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons, Fractals*, vol. 21, no. 3, pp. 749-761, 2004.

[20] C. C. Chang, J. Y. Hsiao and C. L. Chiang, "An image copyright protection scheme based on torus automorphism," in *Proc. of the First International Symposium on Cyber Worlds*, pp. 217-224,

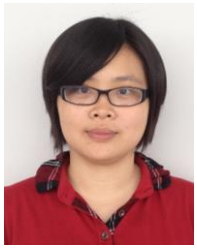
Copyright ⓒ 2013 KSII

2002.
[21] F. Chen, K. W. Wong, X. F. Liao and T. Xiang, "Period distribution of generalized discrete arnold cat map for N = pe," *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 445-452, 2012.
[22] I. Percival and F. Vivaldi, "Arithmetical properties of strongly chaotic motions," *Physica D: Nonlinear Phenomena*, vol. 25, no. 1-3, pp. 105-130, 1987.



Qian Mao was born in Shanxi Province, China, in 1978. She received the B.S. degree in Mechanical Engineering and Automation Science from Nanjing University of Aeronautics and Astronautics, Jiangsu, China, in 2000, and M.E. degrees in Traffic Information Engineering and Control from Shanghai Ship and Shipping Research Institute, Shanghai, China, in 2003, and the Ph.D. degree in Traffic Information Engineering and Control from Tongji University, Shanghai, China, in 2006. Since 2006, she has been with the faculty of the School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, where she is currently a Lecturer. She is also a post-doctoral researcher of Asia University, Taiwan. Her research interests include information security, image processing, and information theory and coding.

Chin-Chen Chang received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Taiwan. He was the head of, and a professor in, the Institute of Computer Science and Information Engineering at the National Chung Cheng University, Chiayi, Taiwan. From August 1992 to July 1995, he was the dean of the College of Engineering at the same university. From August 1995 to October 1997, he was the provost at the National Chung Cheng University. From September 1996 to October 1997, Dr. Chang was the Acting President at the National Chung Cheng University. From July 1998 to June 2000, he was the director of Advisory Office of the Ministry of Education of the R.O.C. Since February 2005, he has been a Chair Professor of Feng Chia University. He is currently a Fellow of IEEE and a Fellow of IEE, UK. He also published several hundred papers in Information Sciences. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.



Hsiao-Ling Wu was born in Kaohsiung, Taiwan, in 1986. She received the BS degree in Applied Mathematics from Feng Chia University, Taichung, Taiwan in 2007. She is currently pursuing her Ph.D. degree in information engineering and computer science from Feng Chia University, Taichung, Taiwan. Her current research interests include electronic commerce, information security, cryptography, and mobile communications.